INT2214: OPERATING SYSTEM CONCEPTS

OS

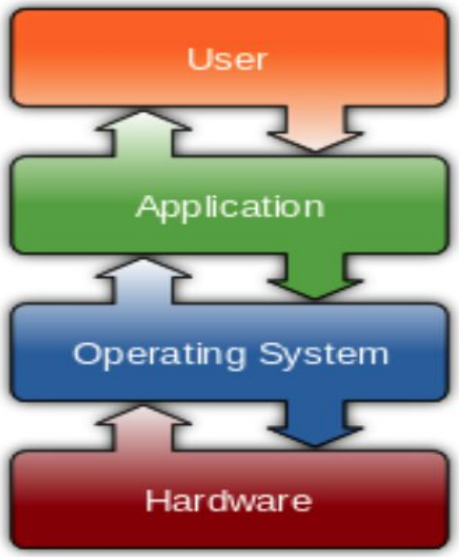
Lesson 1: Introduction

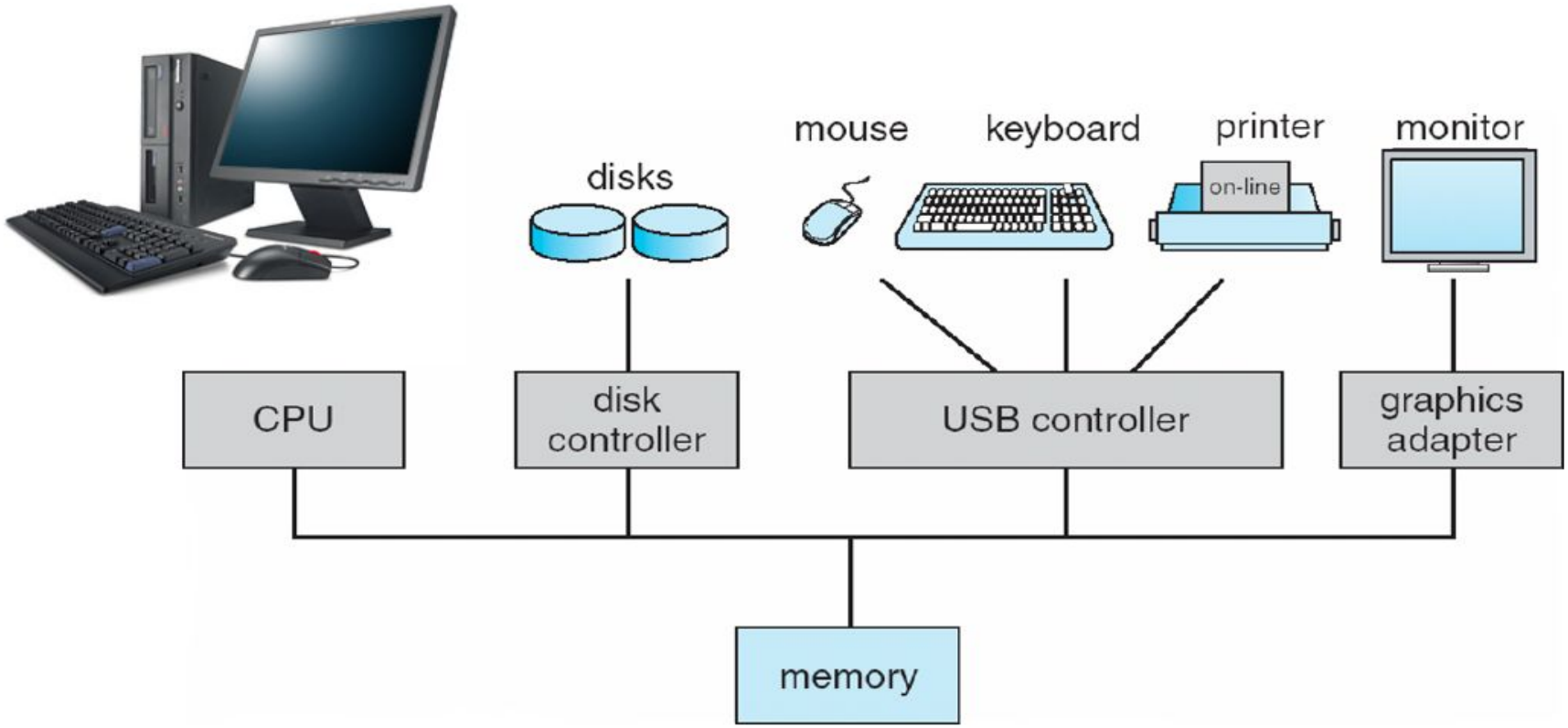
Nguyễn Thị Hậu (UET-VNU)
nguyenhau@vnu.edu.vn

CONTENTS

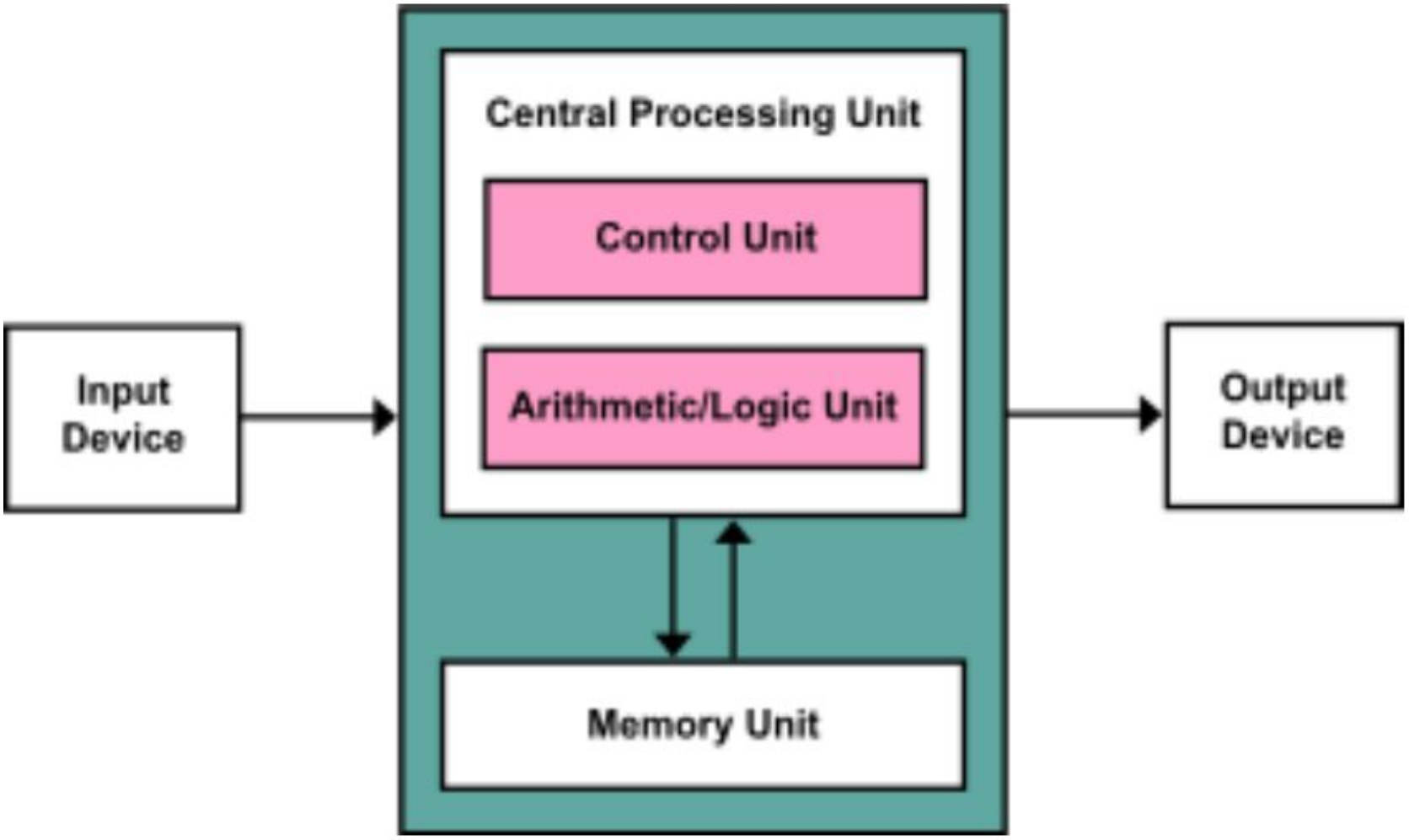
1. Computer-System Organization
2. Computer-System Architecture
3. Operating-System Operations

Components of computer



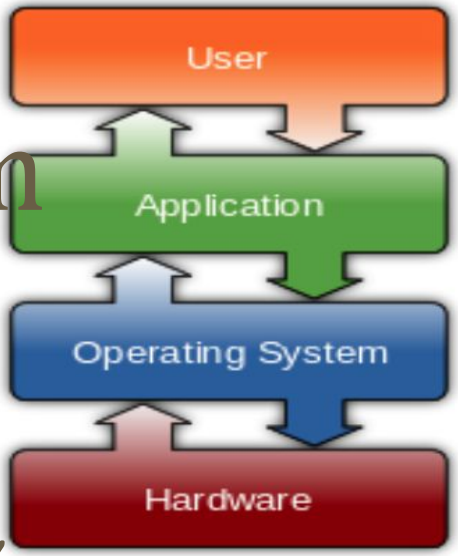


Von Neumann ‘s architecture



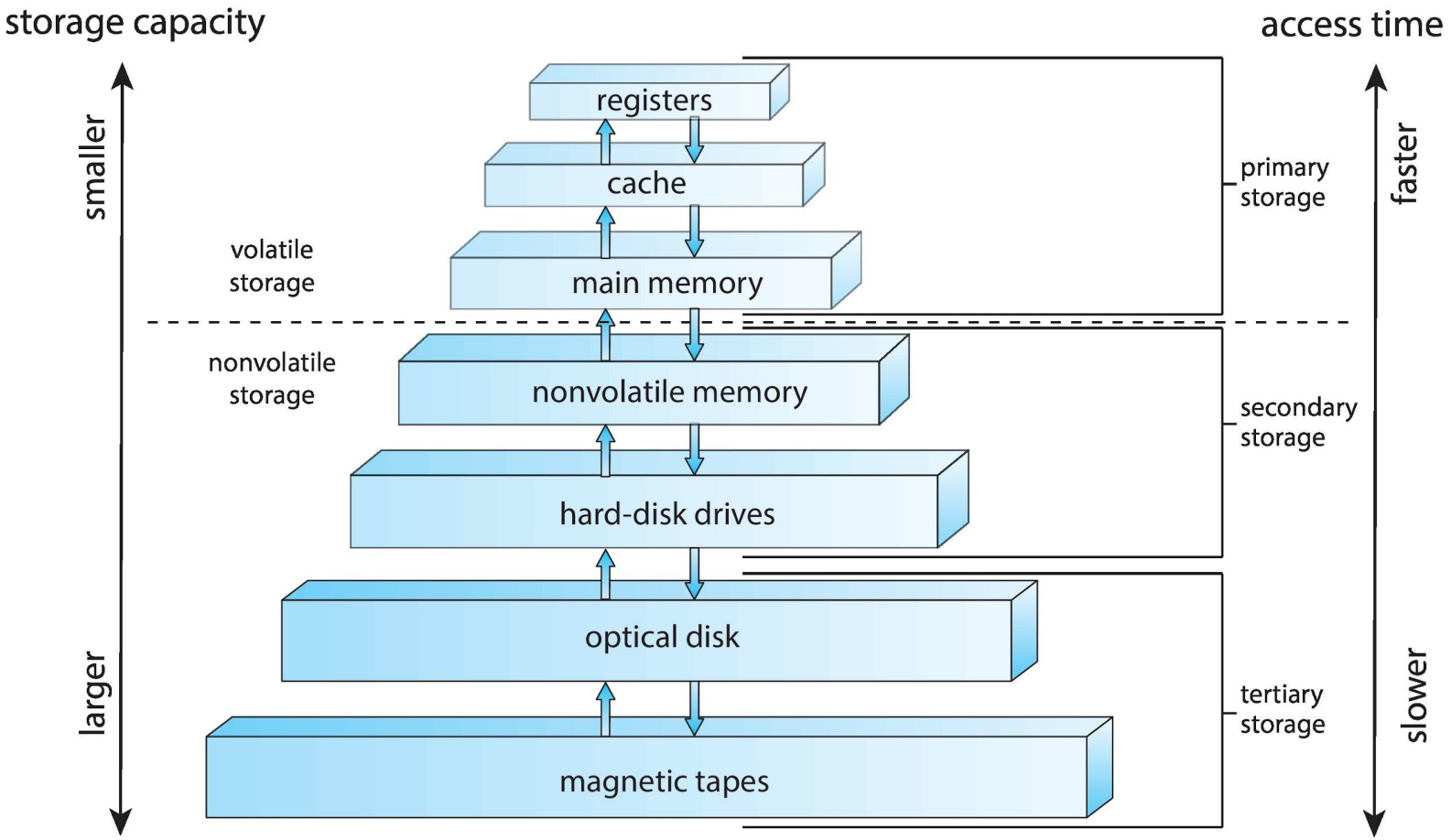
CPU

What is Operating System



- *No universally accepted definition*
- *Operating system is a **resource allocator** and **control program** making efficient use of hardware and managing execution of user programs*
- "The one program running at all times on the computer" is the **kernel,** part of the operating system
- Everything else is either
- a **system program** (ships with the operating system, but not part of the kernel) , or
- an **application program**, all programs not associated with the operating system
- Today's OSes for general purpose and mobile computing also include **middleware** – *a set of software frameworks that provide additional services to application developers such as databases, multimedia, graphics*

Storage-Device Hierarchy



[OS concepts by Silberschatz, Peter Baer Galvin, Greg Gagne 2018]

Comparison of Storage device

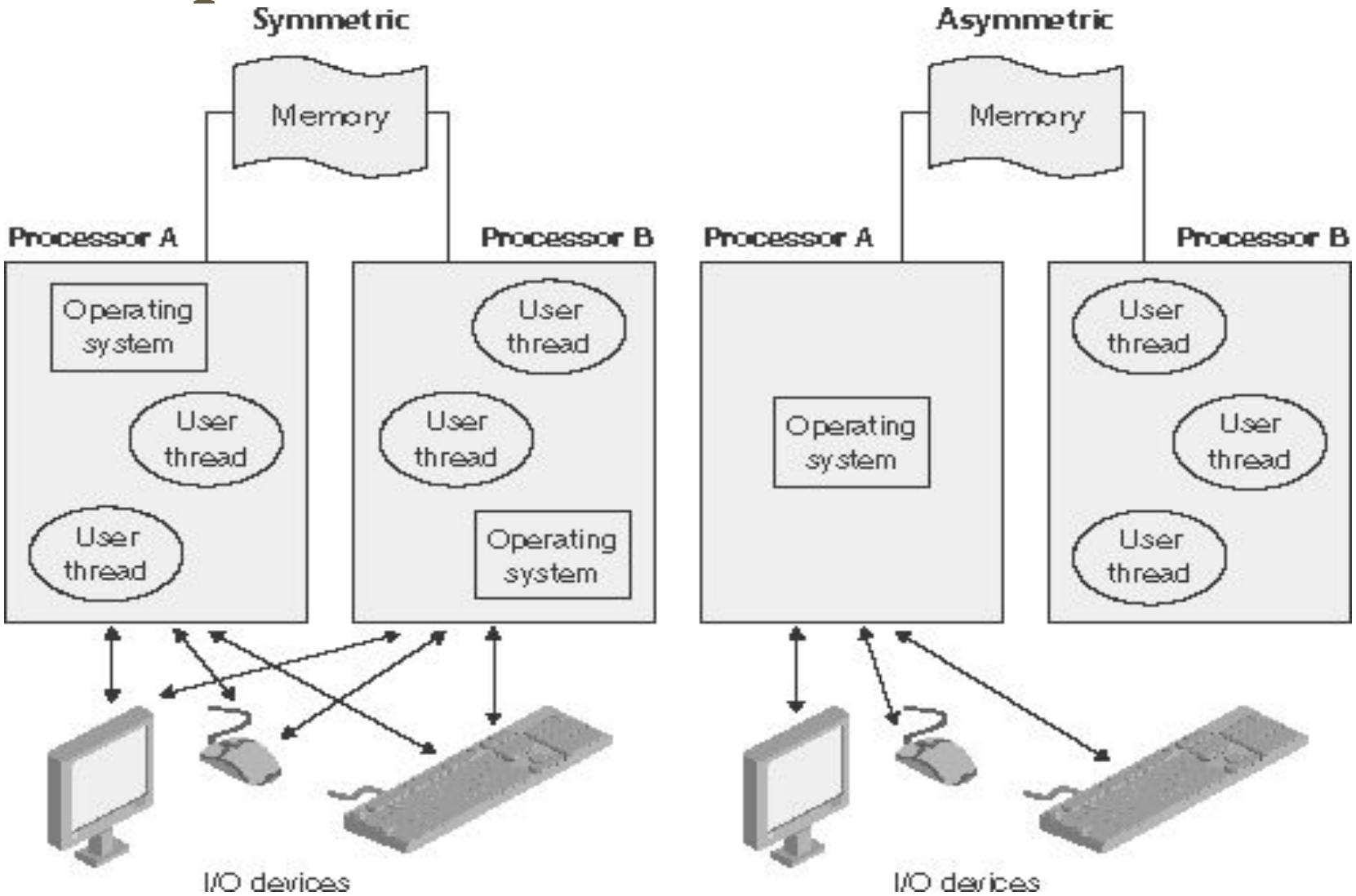
Level	1	2	3	4	5
Name	registers	cache	main memory	solid state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25 - 0.5	0.5 - 25	80 - 250	25,000 - 50,000	5,000,000
Bandwidth (MB/sec)	20,000 - 100,000	5,000 - 10,000	1,000 - 5,000	500	20 - 150
Managed by	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape

[OS concepts by Silberschatz, Peter Baer Galvin, Greg Gagne 2013]

Computer-System Architecture

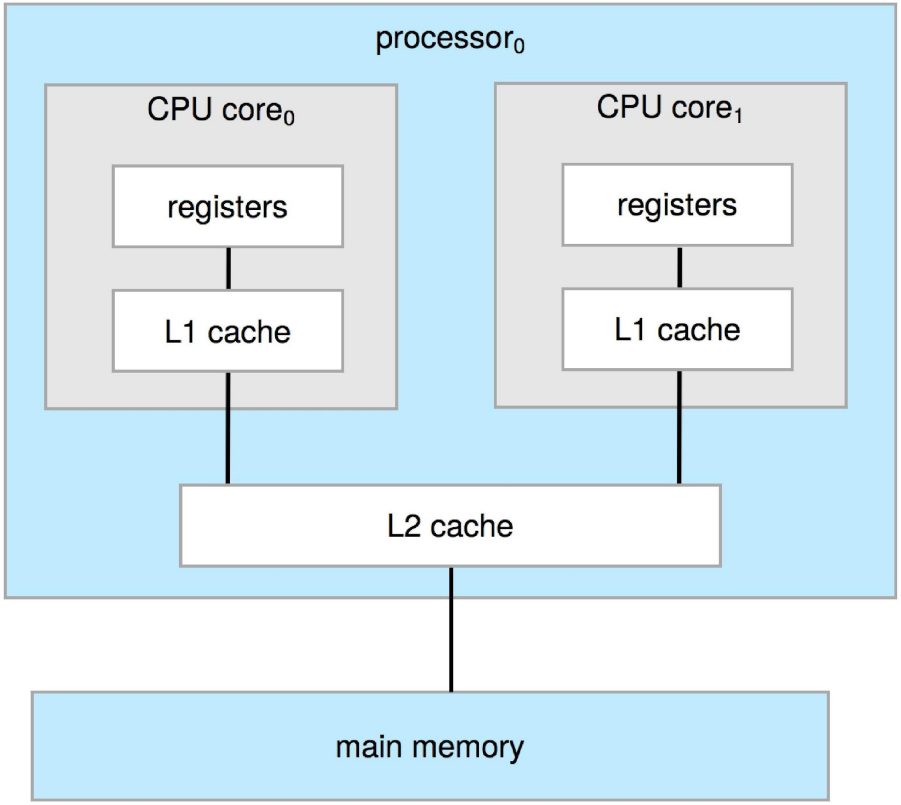
- Most systems use a single general-purpose processor
  - Most systems have special-purpose processors as well
- **Multiprocessors** systems growing in use and importance
  - Also known as **parallel systems**, **tightly-coupled systems**
  - Advantages include:
    1. **Increased throughput**
    2. **Economy of scale**
    3. **Increased reliability** – graceful degradation or fault tolerance
  - Two types:
    1. **Asymmetric Multiprocessing** – each processor is assigned a specie task.
       - **Symmetric Multiprocessing** – each processor performs all tasks

Symmetric vs Asymmetric Multiprocessors

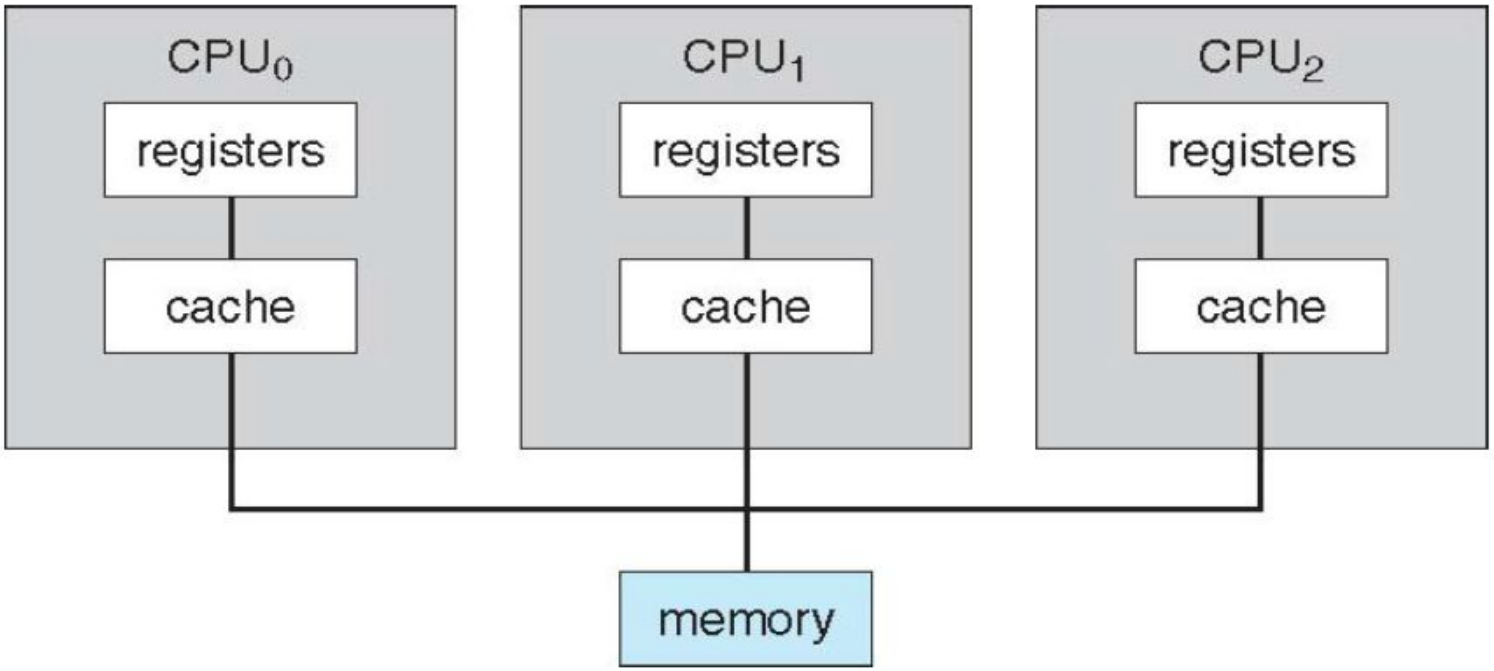


A Dual-Core Design

- Multi-chip and **multicore**
- Systems containing all chips
  - Chassis containing multiple separate systems



Symmetric Multiprocessors

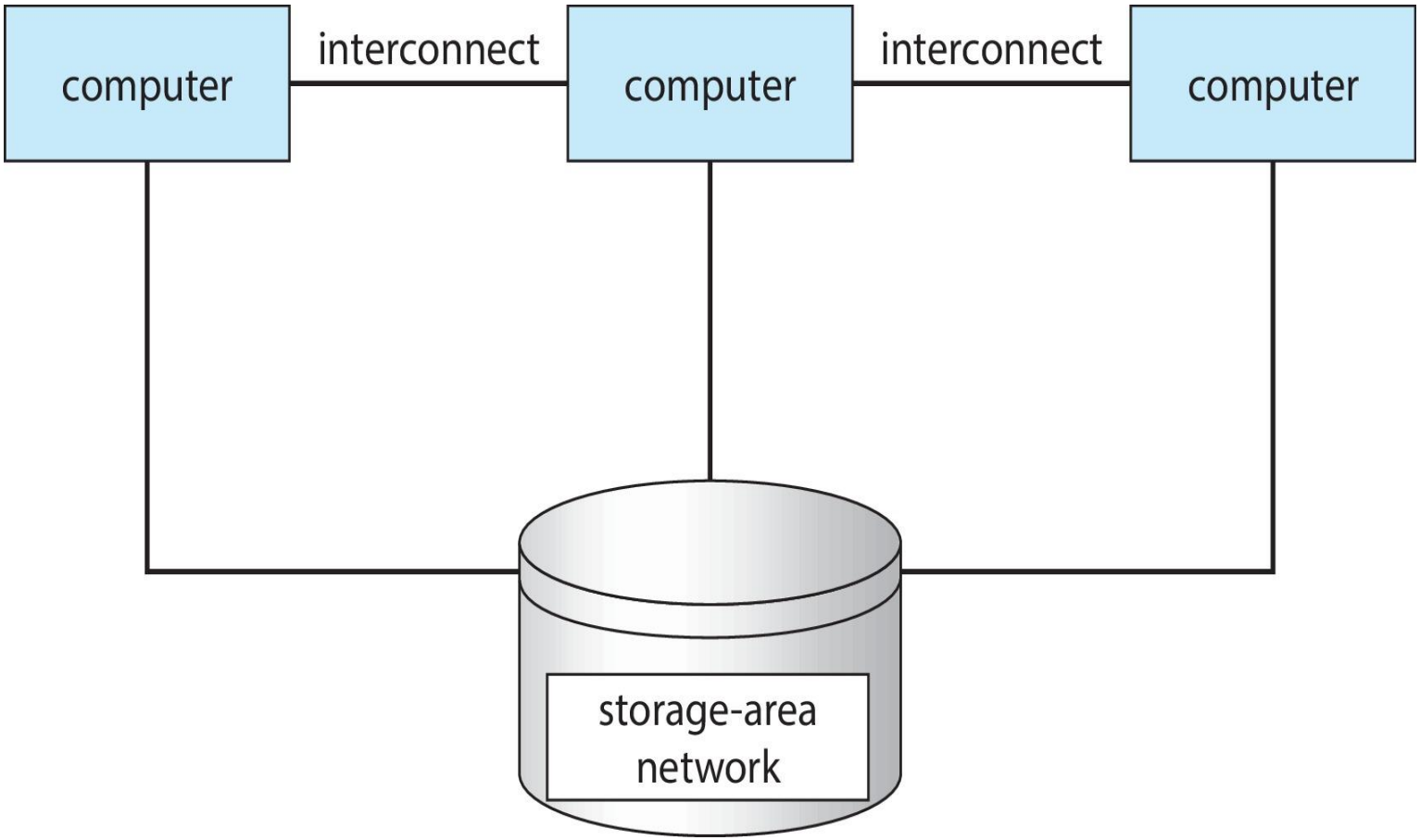


Processor i3 contains 2 cores
Processor i5 contains 2 or 4 cores
Processor i7 contains 2,4, or 6 cores

Clustered Systems

- Like multiprocessor systems, but multiple systems working together
  - Usually sharing storage via a **storage-area network (SAN)**
  - Provides a **high-availability** service which survives failures
    - **Asymmetric clustering** has one machine in hot-standby mode
    - **Symmetric clustering** has multiple nodes running applications, monitoring each other
  - Some clusters are for **high-performance computing (HPC)**
    - Applications must be written to use **parallelization**
  - Some have **distributed lock manager** (**DLM**) to avoid conflicting operations

Clustered Systems



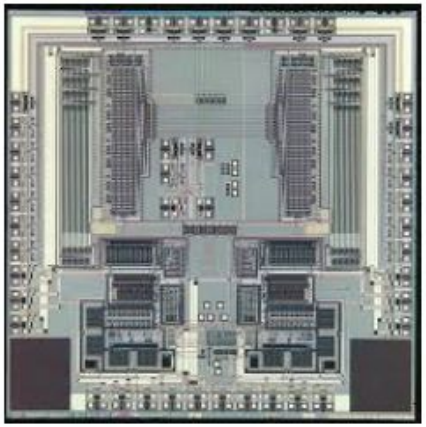
History of computers

Vacuum tube

Thế hệ thứ nhất 1945 – 1955

Transistor

Thế hệ thứ hai 1955 – 1965



Thế hệ thứ ba 1965 – 1980

Thế hệ thứ tư 1980 – nay

Exaflops Computer

- Performance ~ 10^{18} exaflops

1. **Frontier (11/2023), ORNL, USA, 1.194 Exaflop/s, 22.703 MW**
2. Aurora 585 pflops, Intel, 24.686 MW
3. Eagle, 561.20 pflops, Microsoft
4. FUGAKU, 442.01 pflops, Fujitsu, 29.899 MW
5. LUMI, HPE, Finland, 379.7 pflops, 6 MW
6. Leonardo, Atos, 238.7 pflops, 7.404 MW
7. IBM Summit, Mỹ, 148.6 pflops, 10.096 MW

Blue Gene/P supercomputer at Argonne Lab, USA contains 250,000 processor placed in 72 cabin
[Top500.org, Jan 2024]

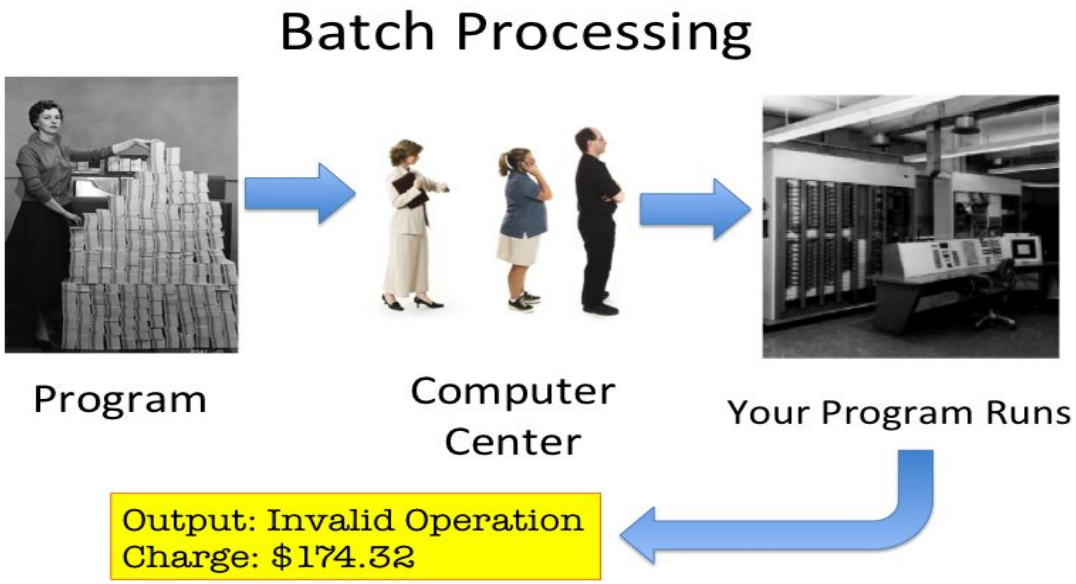
Quantum computing

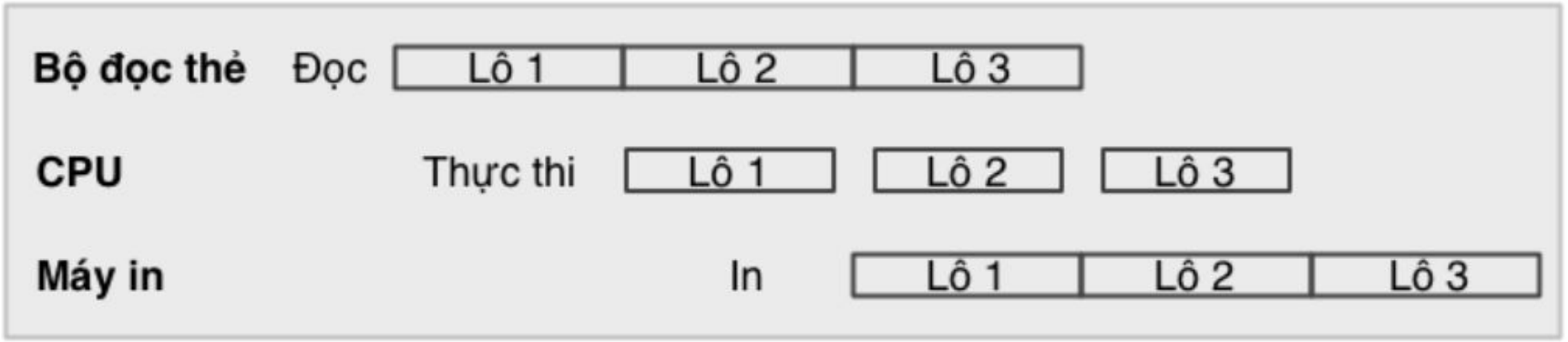
Two qubits can represent four values simultaneously: 00, 01, 10, and 11, again in weighted combinations. Similarly, three qubits can represent 2^3, or eight values simultaneously: 000, 001, 010, 011, 100, 101, 110, 111. Fifty qubits can represent over one quadrillion values simultaneously, and 100 qubits over one quadrillion squared

- 2019: IBM 27 Qubit
- 2020: IBM 65 Qubit
- 2021 IBM 127 Qubit

[https://research.ibm.com/blog/127-qubit-quantum-processor-eagle]

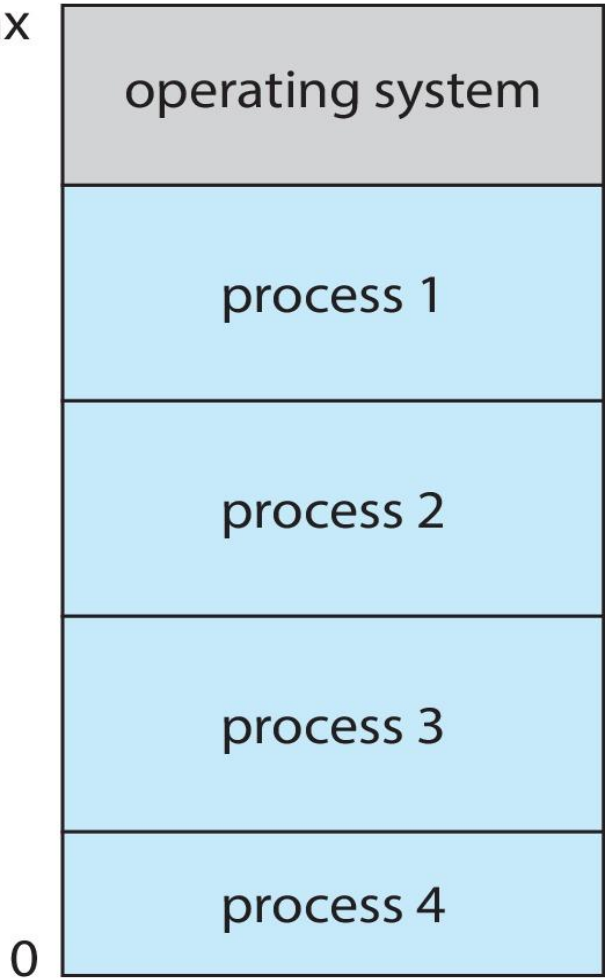
Batch Processing





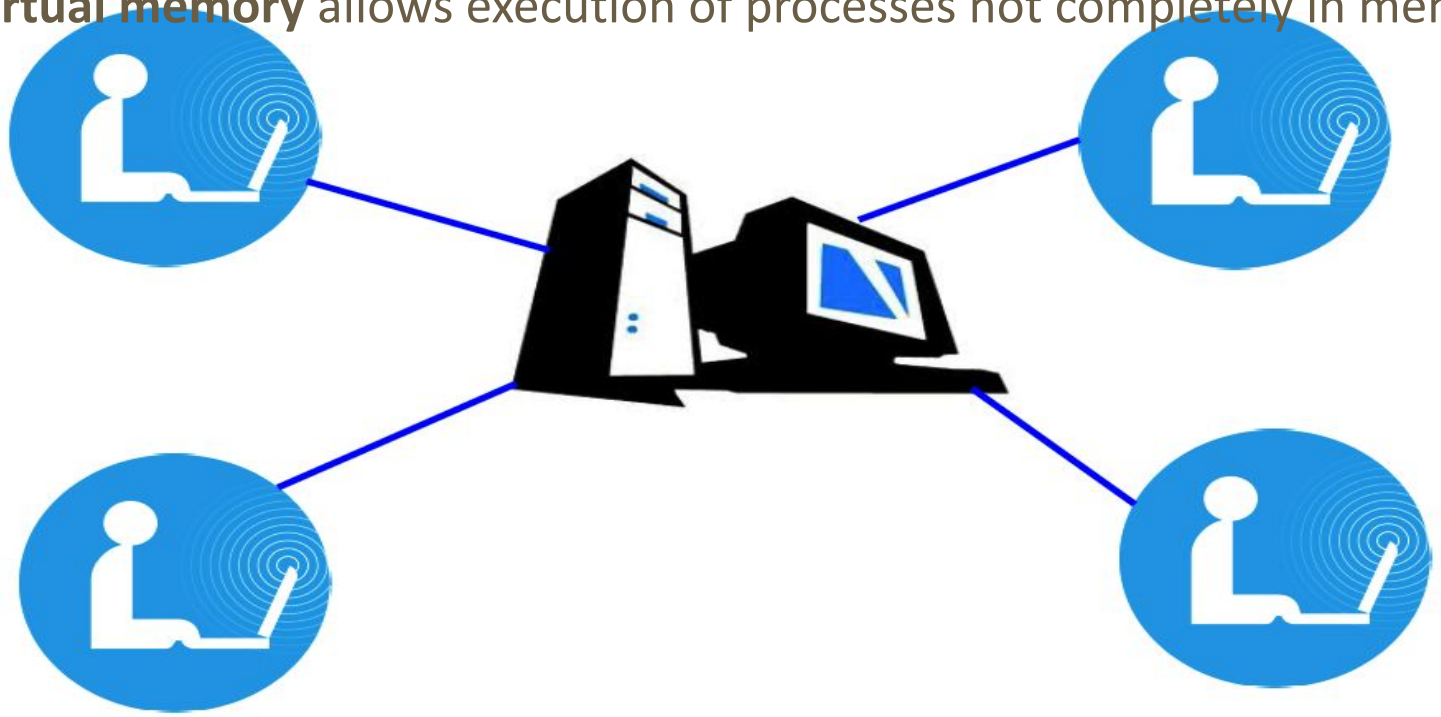
Multiprogramming

- **Multiprogramming** (**Batch system**) needed for efficiency
  - Single user cannot keep CPU and I/O devices busy at all times
  - Multiprogramming organizes jobs (code and data) so CPU always has one to execute
  - A subset of total jobs in system is kept in memory
  - One job selected and run via **job scheduling**
  - When it has to wait (for I/O for example), OS switches to another job



Multitasking/Timesharing

- **Timesharing** (**multitasking**) is logical extension in which CPU switches jobs so frequently that users can interact with each job while it is running, creating **interactive** computing
  - **Response time** should be < 1 second
  - Each user has at least one program executing in memory □ **process**
  - If several jobs ready to run at the same time □ **CPU scheduling**
  - If processes don't fit in memory, **swapping** moves them in and out to run
  - **Virtual memory** allows execution of processes not completely in memory

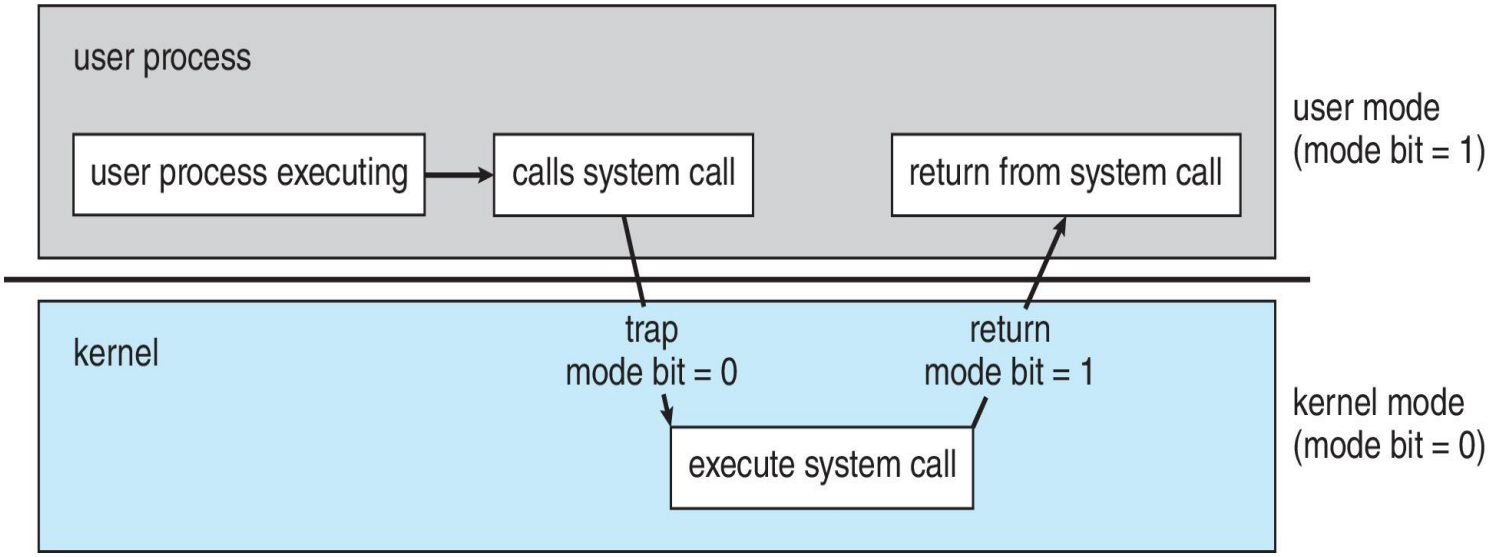


Dual-mode and Multimode Operation

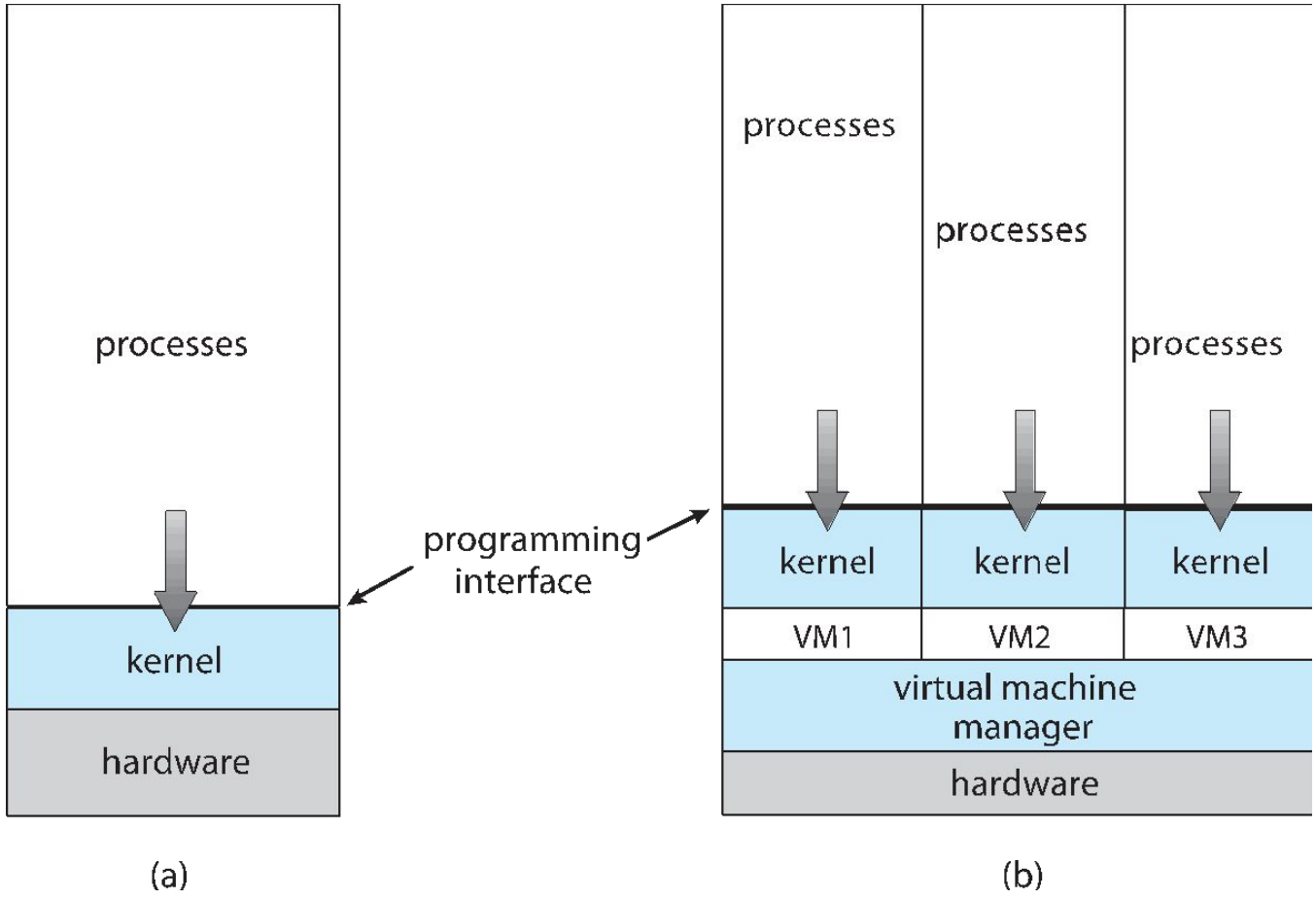
- **Dual-mode** operation allows OS to protect itself and other system components
  - **User mode** and **kernel mode**
  - **Mode bit** provided by hardware
    - Provides ability to distinguish when system is running user code or kernel code
    - Some instructions designated as **privileged**, only executable in kernel mode
    - System call changes mode to kernel, return from call resets it to user
- Increasingly CPUs support multi-mode operations
  - i.e. **virtual machine manager** (**VMM**) mode for guest **VMs**

Transition from User to Kernel Mode

- Timer to prevent infinite loop / process hogging resources
  - Timer is set to interrupt the computer after some time period
  - Keep a counter that is decremented by the physical clock
  - Operating system set the counter (privileged instruction)
  - When counter zero generate an interrupt
  - Set up before scheduling process to regain control or terminate program that exceeds allotted time



Computing Environments - Virtualization




THANK YOU